[WASHING MACHINE PROJECT]

[Washing machine controller for Digital system design]

[ARMEGA ANDREI]

[CRACIUNAS VICTOR]

[30415]

[Semester 2]

[PROJECT SUPERVISOR : ING DIANA POP]

Table of Contents

SECTION 1 : Specifications

Design an automaton that controls a washing machine, with a manual mode and several automatic modes.

Initially, the machine is in an inactive state, with the washing machine door open. The user can set the operating parameters manually (manual mode) or select one of the pre-programmed modes.

In manual mode, you can set: temperature (30°C, 40 °C, 60°C or 90°C); speed (800, 1000, 1200 rotations/minute); select/cancel prewash, additional rinse. The running time of the program depends on the selected temperature (the water comes with a temperature of 15°C and heats up 1°C in 2 seconds) and on the selected function (prewash - same method as the main wash, additional rinse - rinse twice; these functions are described in detail below).

The selectable automatic modes are the following:

- Quick wash - 30°C, 1200 rotations/minute, without prewash, without additional rinsing
- Shirts- 60°C, 800 rotations/minute, without prewash, without additional rinsing
- Dark colours- 40°C, 1000 rotations/minute, without prewash, with additional rinsing
- Dirty laundry- 40°C, 1000 rotations/minute, with prewash, without additional rinsing
- Antiallergic- 90°C, 1200 rotations/minute,without prewash, with additional rinsing

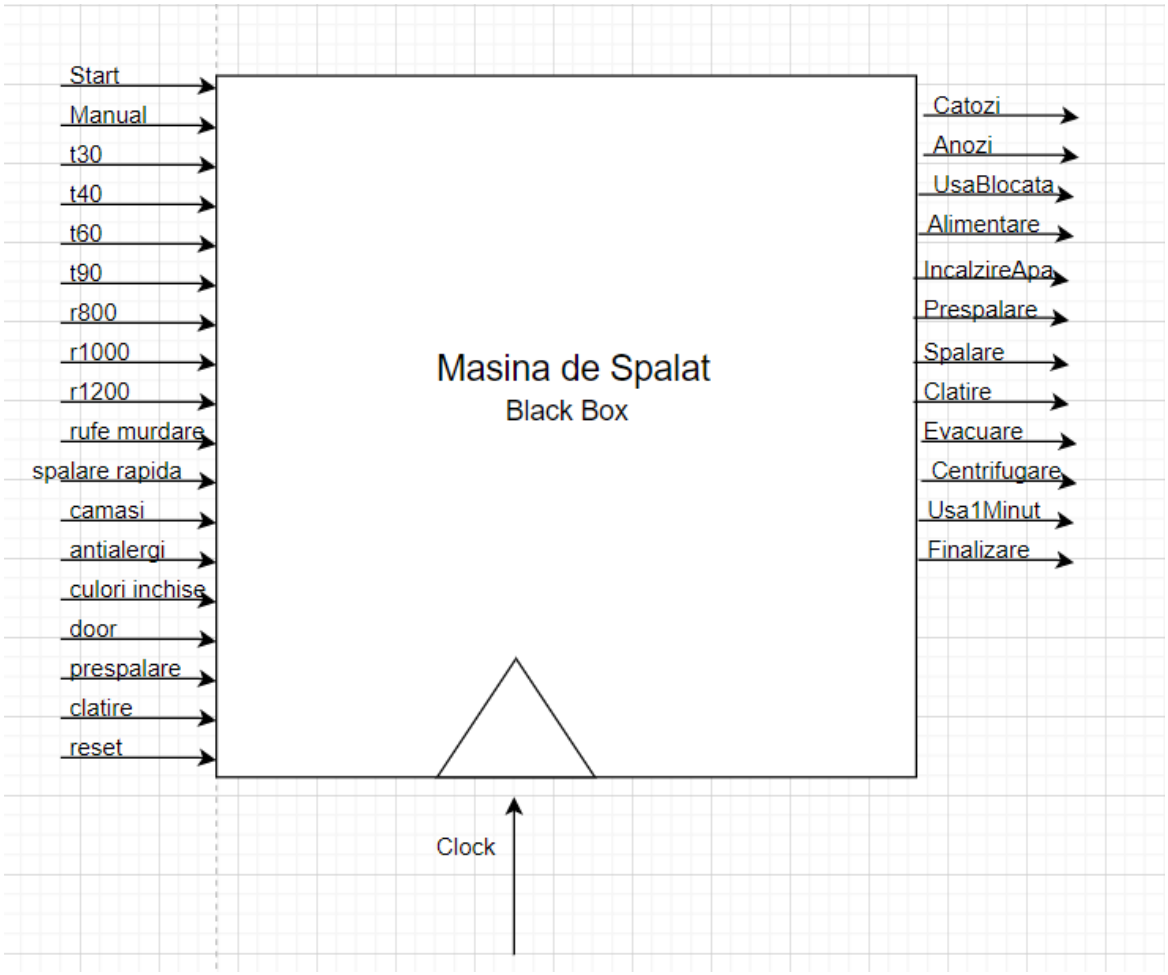
Each program contains the following stages: main wash (fill the machine with water, heat the water, spin at a speed of 60 rotations/minute for 20 minutes, draining) rinse (fill with water, spin at a speed of 120 rotations /minute for 20 minutes, draining) and centrifugation (it rotates at the selected speed for 10 minutes. If the pre-wash is selected, it has the same method as the main wash, except that it rotates for 10 minutes.

The door locks after starting the program and opens one minute after the end of the program. The automaton does not start with the door open.

While selecting the desired mode (manual or one of the automatic modes), the duration of the program is displayed and after it starts, the remaining time is displayed (the time is displayed on 7-segment displays)

SECTION 2: Design

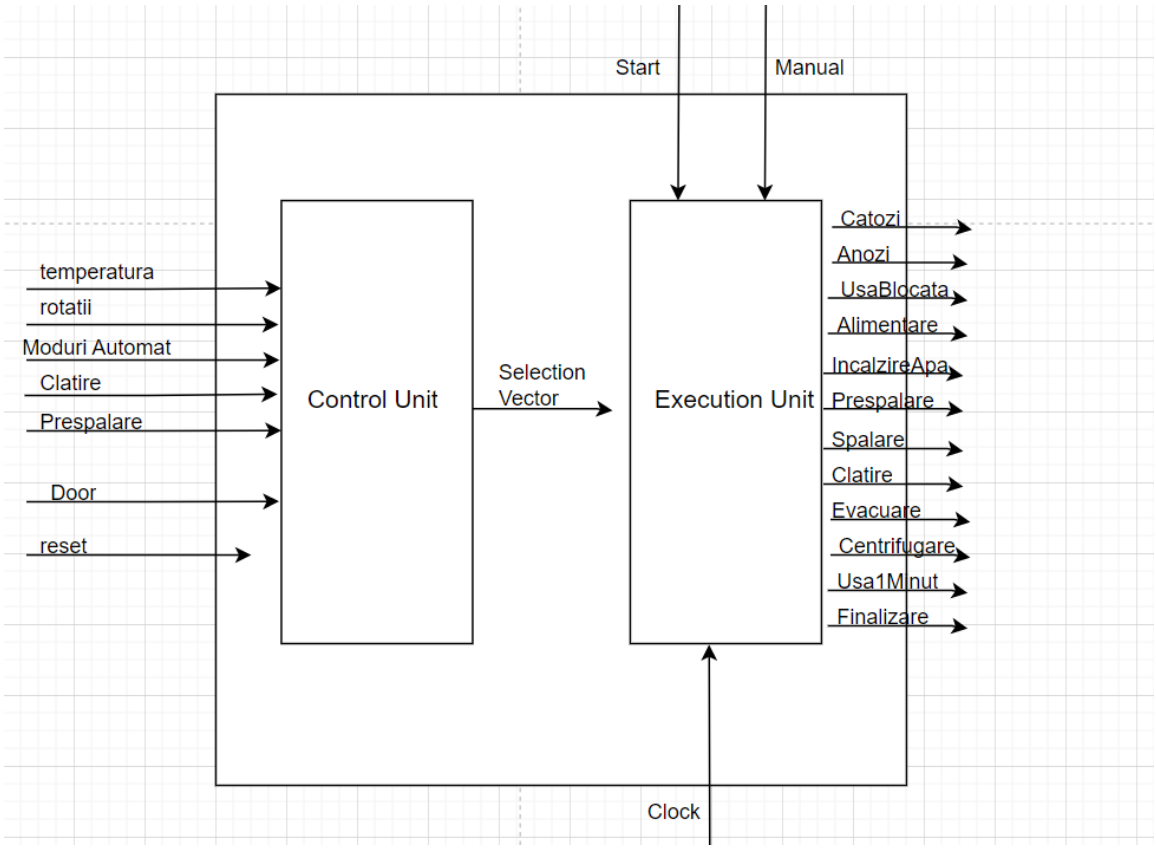
2.1 Black Box



2.1 Control and Execution Unit

The system's black box must be further broken down in order to find implementable components. We will do a top-down breakdown of the problem until we get to known circuits, and then we will implement bottom-up.

2.2.1 Mapping the inputs and outputs of the black box on the two components



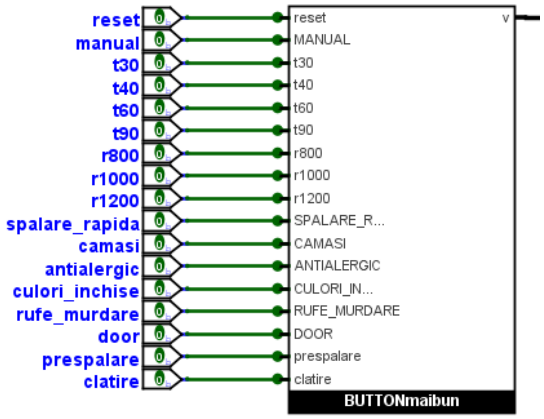
2.2.2 Resources

To further establish the links between the CU and the EU, we must first identify the resources on the basis of which we make decisions. These resources can generate signals to the control unit and can be controlled by the CU via Enable or Reset signals.

Any decision-making information must come from a resource that generates that information and passes it on to UC.

Resources can be simple circuits, which can be implemented directly (counter, register, etc.) or complex resources (remainder algorithm, multiplication algorithm, etc.). These complex resources may appear in the first breakdown with black boxes to which we must establish inputs and outputs, but later they must be further broken down (usually also in CU and EU) until we reach known circuits

1) Buttons



The "buttons" resource works as a vector in which it processes the inputs with the help of and_gates and or_gates. Each input represents a bit in the vector, so the final output will decide the duration of the program. The vector is of length 9 and is constructed as follows:

- V(0)-t30;
- V(1)-t40;
- V(2)-t60;
- V(3)-t90;
- V(4)-r800;
- V(5)-r1000;
- V(6)-r1200;
- V(7)-prewash;
- V(8)-rinse;

2) Starter

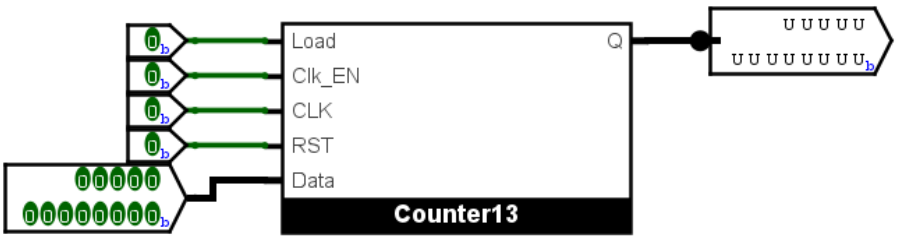
The main purpose of the starter component is to check if the user has entered a predefined automatic program or if he has necessarily entered a certain temperature and rotation (if he has chosen the manual mode). Also, in order for the machine to run, it is necessary that the input for the

door be '1'. Thus, the outputs will be '1' only when the user has satisfied all the requirements of the washing machine and will start the program.

3) Multiplexer 13:1

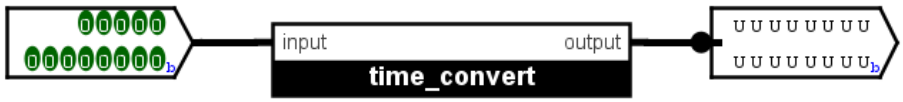
When the user has satisfied all the machine's requirements, the starter will enable the multiplexer, and depending on the modes chosen by the user, it will output the duration of the program based on selection inputs which are on 4 bits. The output is on 13 bits and represents the duration of the program in seconds.

4) Counter



The multiplexer will load the counter with the number of seconds of the program, and it will start counting down to 0. The counter will start counting when the component “starter” will output ‘1’ and the clock will be enabled. Moreover, we provided an asynchronous reset. The clock of this system depends on the user-selectable speed.

5) Time Converter



The time converter receives as input the seconds given by the counter and will convert them into minutes and seconds. The output is on 16 bits because each digit is written in BCD form in order to be displayed.

6) SSD

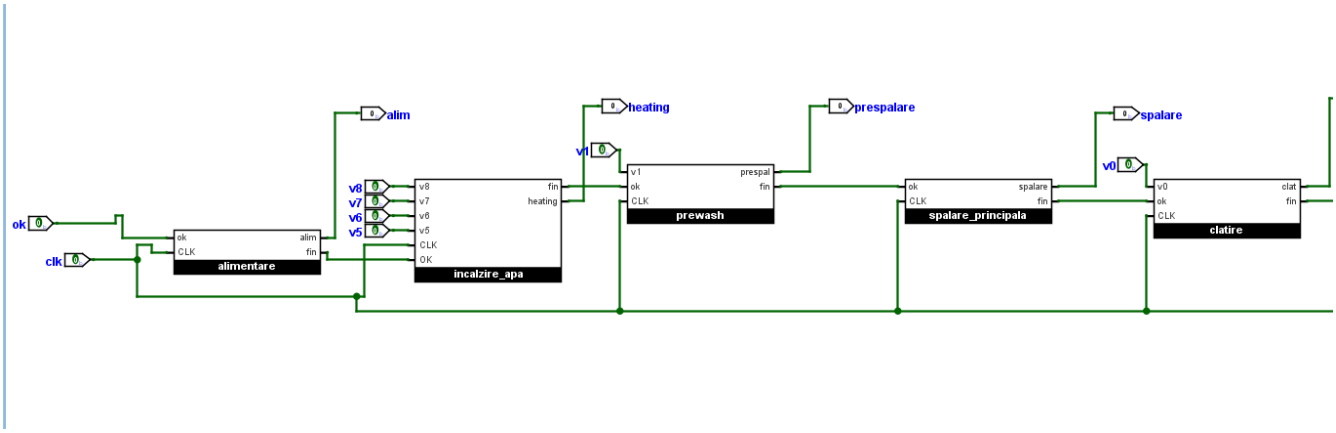
This resource controls the 7-segment display on the board – it displays the time remaining of the program chosen in decimal format. The outputs – cathodes and anodes – are active low signals. Since we have only 7 cathodes (each one corresponding to a segment from *a* to *g*) and we must display a 4-digit decimal number, we will activate the corresponding anodes successively at a great frequency (more than 70 Hz).

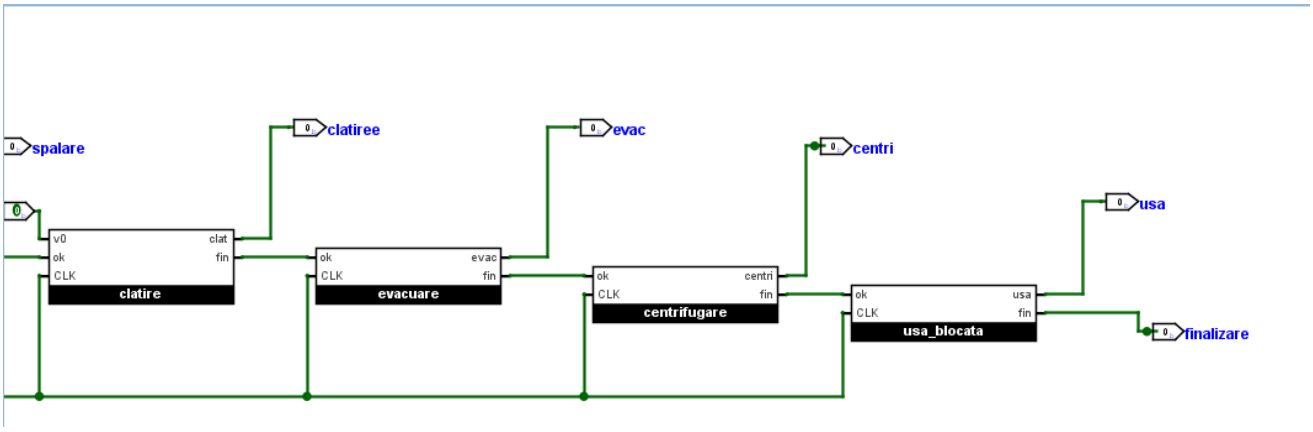
7) Frequency Divider

This resource acts as a frequency divider of the provided clock (100 MHz), bringing the frequency down to 1 Hz, because the initial frequency is too faster for some of the components that we use.

8) Program Masina

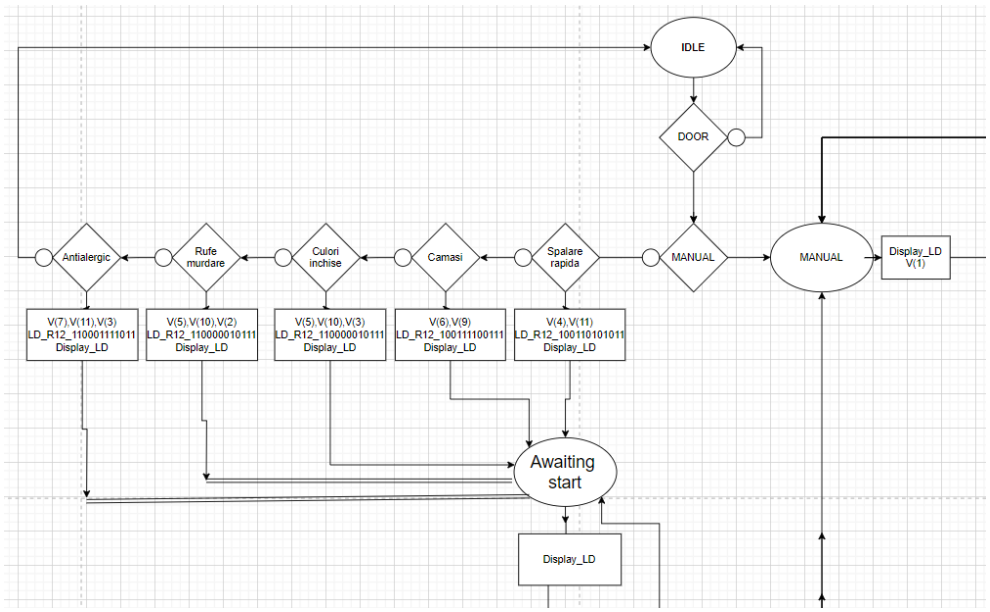
ProgramMasina represents the top-level of several counters. It shows us in real time which program the washing machine is in, giving as output the LED corresponding to the current program. Thus, for each moment of the program there is a counter that counts the minutes for that program. When a program ends, it will enable the next program which also represents a counter. When the program ends, a completion LED will be lit.

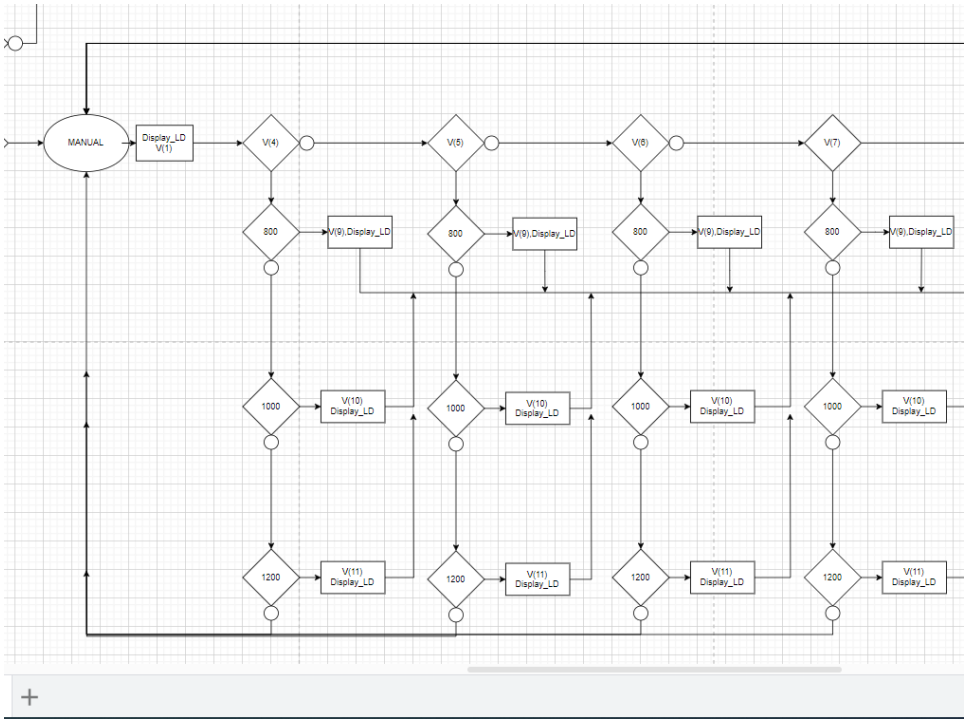


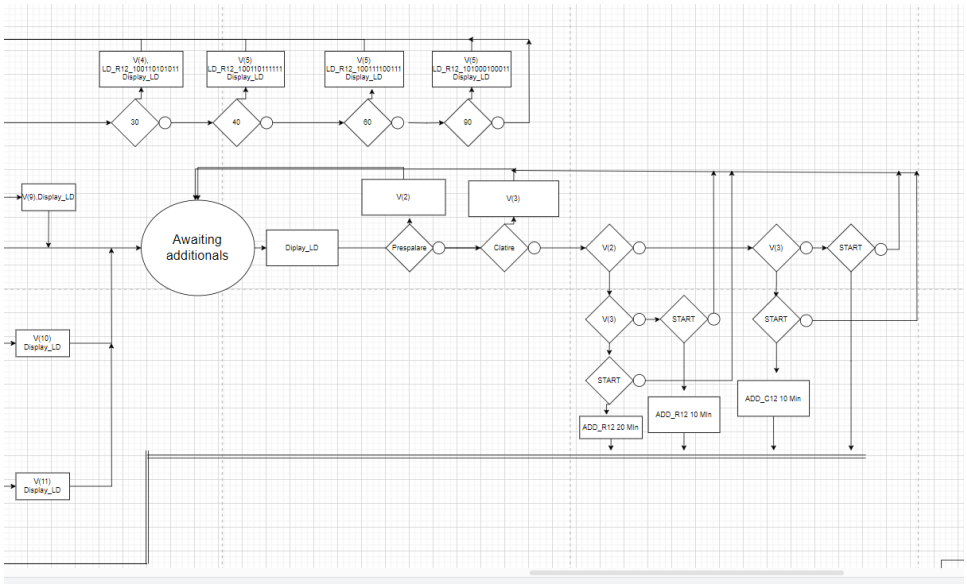


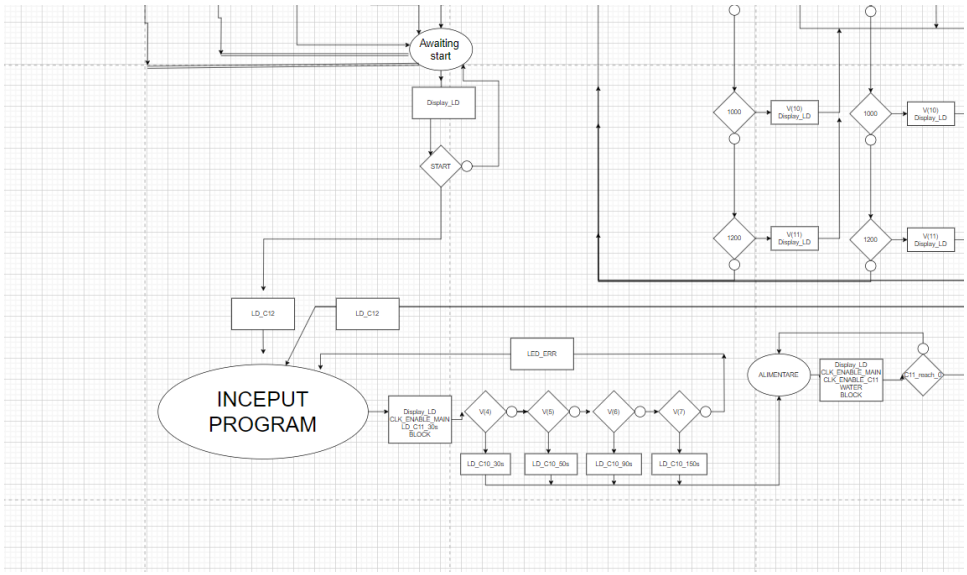
2.2.3 State Diagram

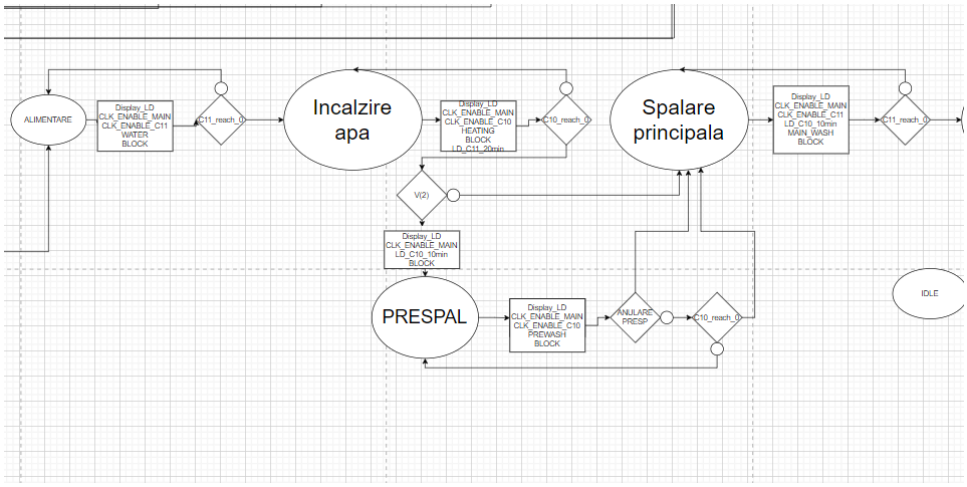
A state represents a moment of time (a period).

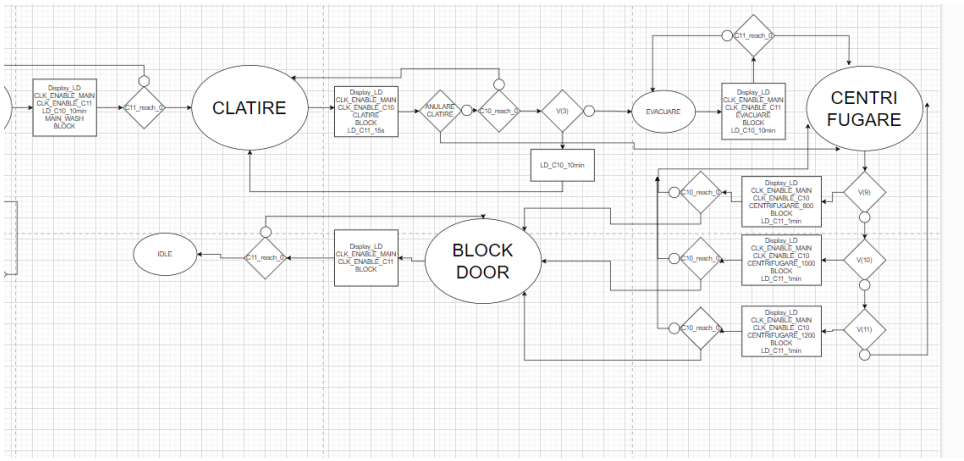

IDLE
DOOR
MANUAL
MANUAL
Display_LD
V(1)
Antialergic
Rufe
murdare
Culori
inchise
Camasi
Spalare
rapida
V(7),V(11),V(3)
LD_R12_110001111011
Display_LD
V(5),V(10),V(2)
LD_R12_110000010111
Display_LD
V(5),V(10),V(3)
LD_R12_110000010111
Display_LD
V(6),V(9)
LD_R12_100111100111
Display_LD
V(4),V(11)
LD_R12_100110101011
Display_LD
Awaiting
start
Display_LD


MANUAL
Display_LD
V(1)
V(4)
V(5)
V(6)
V(7)
800
V(9),Display_LD
1000
V(10)
Display_LD
1200
V(11)
Display_LD

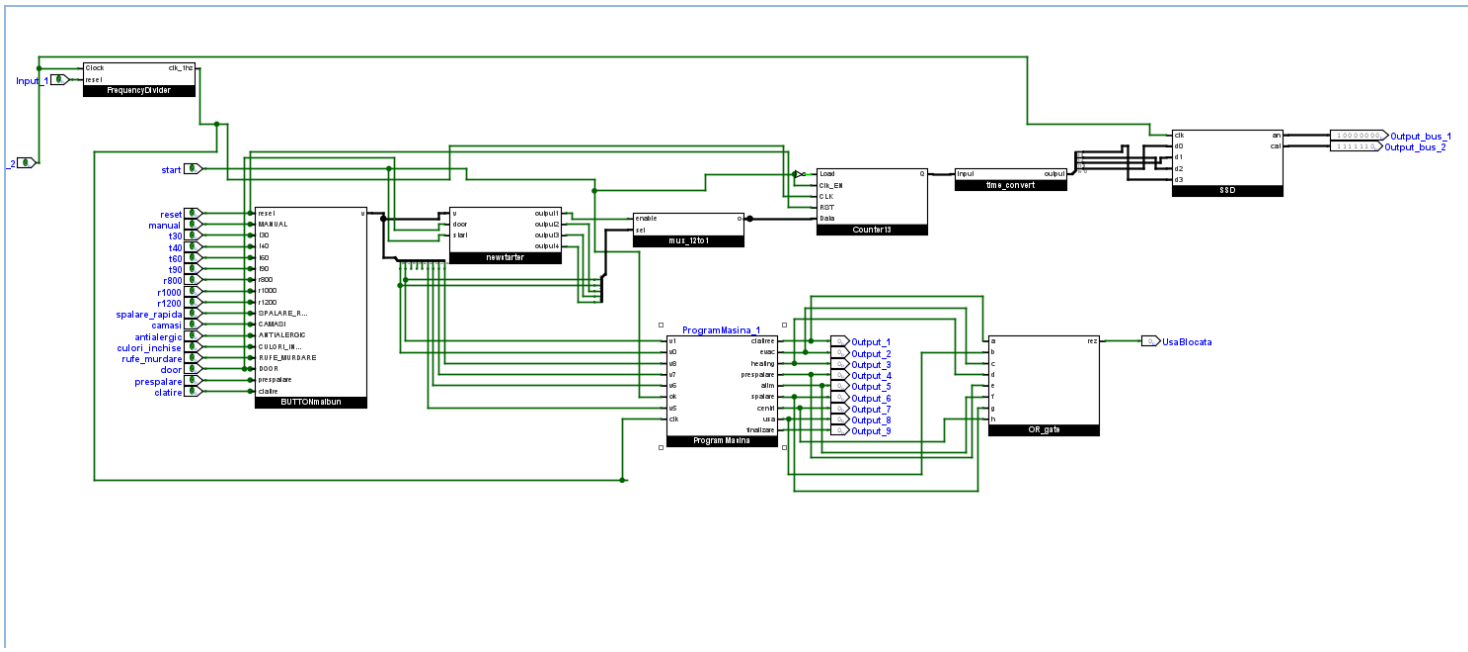

V(4)
LD_R12_100110101011
Display_LD
V(5)
LD_R12_100110111111
Display_LD
V(5)
LD_R12_100111100111
Display_LD
V(5)
LD_R12_101000100011
Display_LD
30
40
80
90
V(9).Display_LD
Awaiting additionals
Display_LD
V(2)
V(3)
Prespalare
Clatire
V(2)
V(3)
START
V(3)
START
START
START
ADD_R12 20 Min
ADD_R12 10 Min
ADD_C12 10 Min
V(10)
Display_LD
V(11)
Display_LD


Awaiting start
Display_LD
START
1000
V(10)
Display_LD
1000
V(10)
Display_LD
1200
V(11)
Display_LD
1200
V(11)
Display_LD
LD_C12
LD_C12
LED_ERR
INCEPUT PROGRAM
Display_LD
CLK_ENABLE_MAIN
LD_C11_30s
BLOCK
V(4)
V(5)
V(6)
V(7)
LD_C10_30s
LD_C10_60s
LD_C10_90s
LD_C10_150s
ALIMENTARE
Display_LD
CLK_ENABLE_MAIN
CLK_ENABLE_C11
WATER
BLOCK
C11_reach_0


ALIMENTARE
Display_LD
CLK_ENABLE_MAIN
CLK_ENABLE_C11
WATER
BLOCK
C11_reach_0
Incalzire apa
Display_LD
CLK_ENABLE_MAIN
CLK_ENABLE_C10
HEATING
BLOCK
LD_C11_20min
C10_reach_0
Spalare principala
Display_LD
CLK_ENABLE_MAIN
CLK_ENABLE_C11
LD_C10_10min
MAIN_WASH
BLOCK
C11_reach_0
V(2)
Display_LD
CLK_ENABLE_MAIN
LD_C10_10min
BLOCK
PRESPAL
Display_LD
CLK_ENABLE_MAIN
CLK_ENABLE_C10
PREWASH
BLOCK
ANULARE PRESP
C10_reach_0
IDLE



2.2.4 Detailed diagram of the project



SECTION 3: User manual

MANUAL MODE

- The washing machine lets you select the temperature, rotations per minute and a prewash or additional rinse .

- **For choosing the specifications of the washing program yourself , the user must press the MANUAL SWITCH.**
- The 4 possible temperatures are 30 ° , 40 ° , 60 °, 90 °; the 3 possible rotations are 800 , 1000 ,1200; the prewashing and additional rinsing options have their 2 separate switches respectively.
- The sequence for choosing the specifications for the manual program are:
- Input the clothes in the washing machine , close the door(switch ; door is close on '1') , press the MANUAL button , then select temperature , then select rotations , then , in no order, you may or may not press PREWASHING SWITCH or ADDITIONAL RINSE SWITCH. If you feel the inputs are not appropriate , flip the switch back and select the right one.
- **After all these selections have been made you may flip the START SWITCH and the washing program will start!**
- **You may NOT start without a temperature and a number of rotations !**
- **You may NOT press an predetermined program while MANUAL switch is on!**
- **You may not turn off the MANUAL SWITCH until START SWITCH has been flipped**
- **If you flip back the START switch while the program will stop**
- **The program will not start if the MANUAL and DOOR switches are off!!!**

AUTOMATED MODE

- The washing machine also lets you choose between one of 5 predetermined programs that vary in temperature and rotations; prewashing and additional rinsing may or may not be present. For choosing one of the predefined programs, the use must simply press the name of the program itself.
- **If the user wishes to select an automated program , the manual button must not be pressed!**
- The five predetermined programs are :
- *Spalare rapida :*30° 1200 rot
- *Camasi :* 60° 800 rot
- *Antialergic :*90° 1200 rot additional rinse
- *Culori Inchise:* 40° 1000 rot additional rinse
- *Rufe murdare :*40° 1000 rot prewash
- The Sequence of inputs is : input your clothes in the washing machine , flip the DOOR switch , then select one of the five programs by flipping the apropriate switch and then flip the START SWITCH.
- **ANY OF THE TEMPERATURE ,ROTATION , PREWASH , ADDITIONAL RINSE SWITCHES WILL NOT HAVE EFFECT!!**

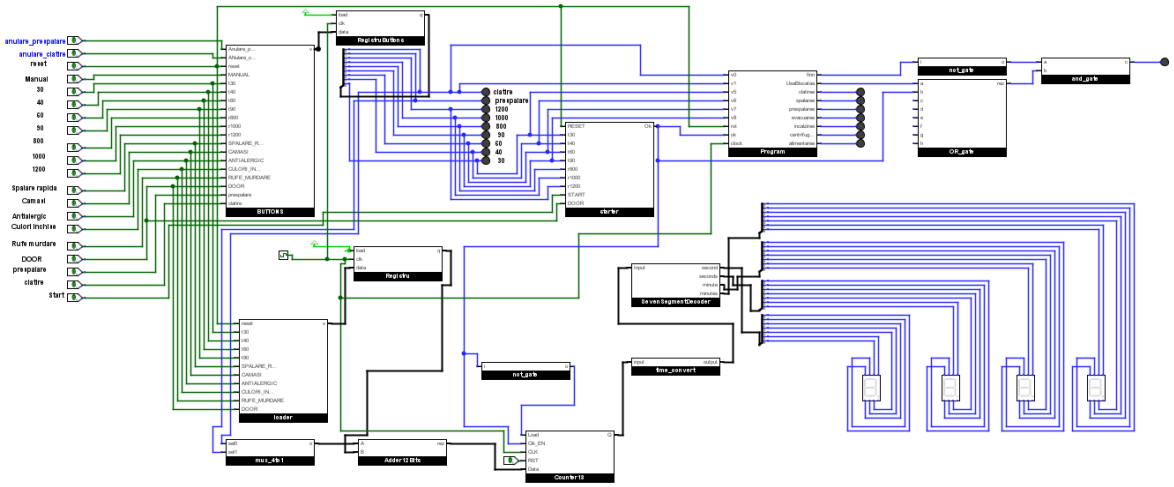
UNWANTED BEHAVIOUR PREVENTION

- **The user shall not have multiple temperatures / rotations / predetermined programs at the same time.**
- **If the user wishes to stop the program after it started , flip to 0 all switches and then flip the RESET switch**
- **Do not use the RESET switch in combination with other switches**

SECTION 4 : TECHINCAL JUSTIFICATIONS

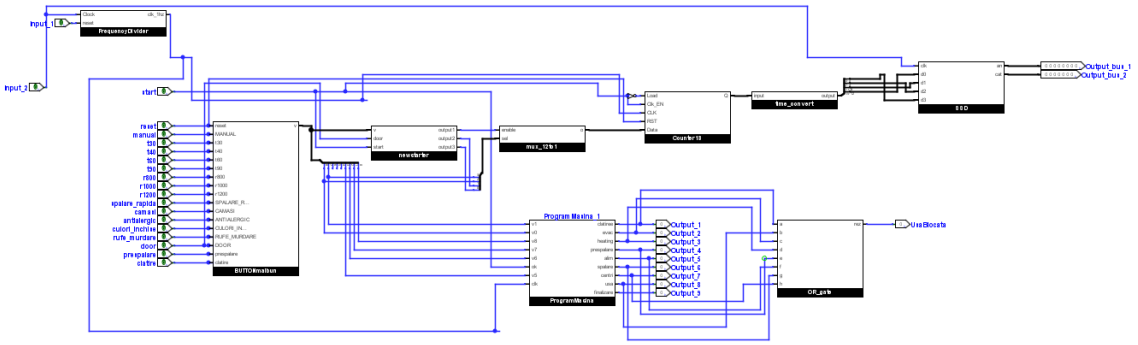
- The washing machine has been designed in two iterations .

  Iteration one:

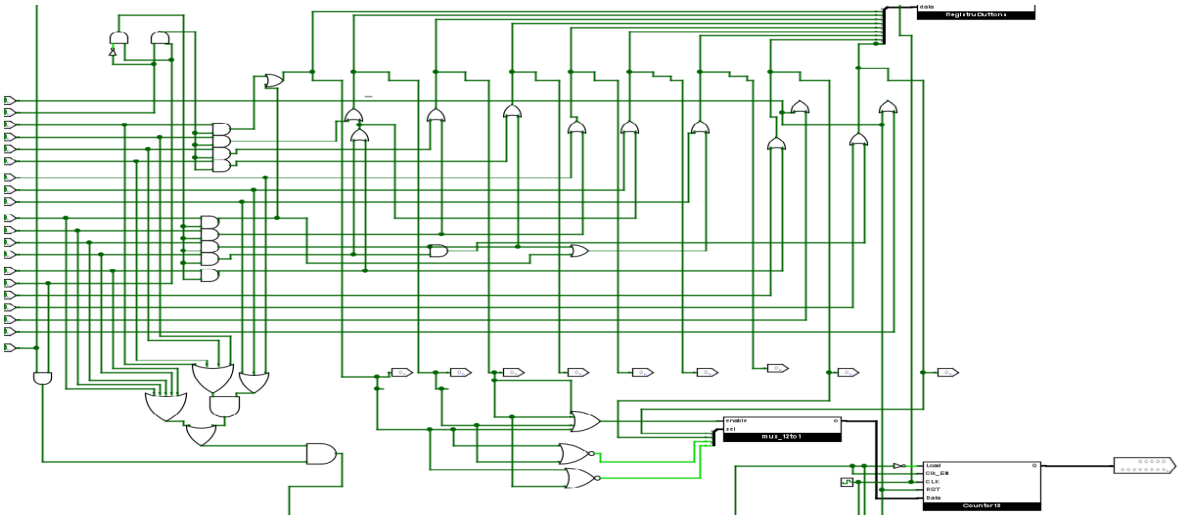


- This iteration works by taking the input selections and storing each possible value for the program and storing it into a register . The time for the program is calculated dynamically by adding the hardwired values from the component LOADER and another 3 values from the 4 to 1 multiplexer. The sum is loaded dynamically into the counter that counts the main program.

Second one :



- This iteration has the same functionality, but the components have been reduced .
- This iteration opts to not store the values in a register for the sake of easier user interaction . Furthermore , the LOADER and ADDER components have been removed , all the inputs being calculated into selections for a 12:1 multiplexer , that loads the corresponding time into the counter.
- The button component now calculates the conditions for the program to start , which was the function of the starter component in the previous iteration.



- JUSTIFICATIONS :

- The Buttons , and starter component can be seen in the diagram above. We chose to implement these conditional components by using as little logic gates as possible, and the code is written with logical expressions.
- Instead of having an 13 bit adder and a Multiplexer , we opted for enlarging the multiplexer and calculating the selection with a few simple logic gates.
- The previous starter function is now in BUTTONS component implemented with logic gates. We don't consider this can be done easier than this , in terms of calculating the total time.
- We use a standard frequency divider and seven segment display.
- For the washing program we are cascading counters for all the possible states of the washing program. This could be done with one counter but considering all the complications of possible state skips and led selection , many additional components would be necessary , so we decided that a counter for each state is simpler.
- The washing program concept remains unchanged through both iterations of the washing machine.
- The time converter component is just a series of arithmetic operations that can not be overly simplified so we decided that standard functions should suffice .

SECTION 5 : FUTURE DEVELOPMENTS

- There are a few developments that may be implemented .
- First of all , the 12 :1 multiplexer has 5 select bits , and many of these select values are not being used , so a few other temperatures may be added.
- Second of all , the MAIN counter is on 13 bits , and uses only 12 of those . This additional bit can be used to count much more time than possible at the moment.
- The RESET switch may become obsolete altogether and the START and STOP of the program may be implemented without having the need to RESET the whole program.
- The user inputs can be stored in a REGISTER , and the SWITCHES can be replaced with buttons , but that would require the RESET switch to be present again.
- More unpredictable behavior can be eliminated through further checks on the signals.

SECTION 6 : REFERENCES

- Digital System Design – lectures & laboratories
- Logic Design – lectures & laboratories
- Octavian Creț, Lucia Văcariu, "Logic Design Problems -for digital systems-", U. T. Press, Cluj-Napoca, 2013
- Octavian Creț, Lucia Văcariu, "Limbaj de programare VHDL"